

Universidad de Guadalajara  
Centro Universitario de Ciencias Exactas e Ingenierías



**Curso:** Laboratorio abierto: diseño

**Clave:** IL372

**NRC:** 227976

**Calendario:** 2025-B

**Sección:** D05

**Profesor:** LUIS ERNESTO AYALA HERNANDEZ

**Actividad:** Informe de diseño final

**Alumno:** Castillo Rocha Bryan Alejandro

**Código Alumno:** 218456605

## **1. Introducción**

Este informe consolida el diseño integral del proyecto "Plataforma de Descubrimiento Local Inteligente", desarrollado en el curso de Laboratorio Abierto: Diseño. El objetivo principal de este documento es presentar de manera coherente y profesional el proceso de diseño seguido, desde la conceptualización del problema hasta la planificación detallada para su implementación.

El proyecto aborda la "doble brecha" existente en el ecosistema de ocio local: la dificultad de los usuarios para encontrar "joyas ocultas" y la limitada visibilidad de los negocios emergentes. La solución propuesta es una plataforma móvil inteligente que utiliza un motor de recomendación con IA para conectar eficientemente a ambos lados del mercado.

A lo largo de este documento, se detallan las fases clave del diseño, incluyendo la definición de requerimientos, el análisis de factibilidad, la evaluación de alternativas, el diseño de la arquitectura y la planificación final del proyecto. Este informe representa la culminación de dicho proceso y sirve como el plano maestro para el desarrollo futuro de la plataforma.

## **2. Documento Inicial: Problema y Objetivos**

### **2.1. Descripción del Problema**

El problema central que se aborda es una doble brecha en el ecosistema de ocio local.

- **Para los usuarios:** Enfrentan una sobrecarga de información y una búsqueda fragmentada al intentar responder "¿Qué plan hacemos hoy?". La dependencia de recomendaciones genéricas dificulta el descubrimiento de actividades que se alineen genuinamente con sus gustos, presupuesto o contexto.
- **Para los negocios locales:** Los negocios emergentes, eventos y actividades culturales pequeñas sufren de visibilidad muy limitada, compitiendo en un entorno digital dominado por grandes presupuestos publicitarios.

Resolver esta desconexión fomenta la economía local y democratiza el descubrimiento basado en la afinidad.

### **2.2. Objetivo General**

Desarrollar una plataforma móvil de descubrimiento local inteligente que, mediante un motor de recomendación con IA, conecte de manera eficiente a los usuarios con planes, actividades y lugares hiperlocales personalizados, al mismo tiempo que ofrece a los negocios emergentes un canal de marketing directo para alcanzar a su público objetivo.

### **2.3. Objetivos Específicos**

1. Diseñar y construir una interfaz de usuario intuitiva que permita a los usuarios filtrar opciones de manera contextual (presupuesto, compañía, ambiente) y que aprenda de sus interacciones.
2. Implementar un algoritmo de recomendación proactivo (MVP) capaz de sugerir "joyas ocultas" con alta probabilidad de ser del agrado del usuario, emulando el "Efecto Spotify".
3. Crear un portal web autogestionable para negocios y organizadores donde puedan registrar sus locales y eventos, asegurando una base de datos diversa.

## **3. Documento de Requerimientos**

### **3.1. Requerimientos Funcionales (RF)**

Describen las acciones específicas que el sistema debe realizar.

- **RF-1 (Registro y Autenticación de Usuarios):** Permitir registro vía correo y redes sociales.
- **RF-3 (Búsqueda y Filtrado Contextual):** Filtros por presupuesto, compañía, ambiente, tipo de actividad.
- **RF-5 (Feed de Descubrimiento Proactivo):** Pantalla principal con recomendaciones personalizadas por IA ("joyas ocultas").
- **RF-9 (Gestión de Perfil de Negocio):** Portal web autogestionable para que las empresas administren su perfil, fotos, horarios, etc.

### **3.2. Requerimientos No Funcionales (RNF)**

Definen las características de calidad y restricciones del sistema.

- **RNF-1 (Usabilidad):** Interfaz intuitiva tanto en la app móvil como en el portal de negocios.
- **RNF-2 (Rendimiento):** Tiempos de carga para el feed y búsquedas no deben exceder los 3 segundos.
- **RNF-3 (Seguridad):** Datos sensibles encriptados y comunicaciones vía HTTPS.
- **RNF-6 (Escalabilidad):** La arquitectura debe soportar un crecimiento exponencial de usuarios y negocios.

### **3.3. Requerimientos Críticos**

Los tres requerimientos indispensables para el éxito del proyecto:

1. **RF-5: Feed de Descubrimiento Proactivo:** Es el núcleo de la propuesta de valor y el principal diferenciador.
2. **RF-9: Gestión de Perfil de Negocio:** El contenido de la plataforma depende de la participación de los negocios; un portal autogestionable es vital.
3. **RNF-1: Usabilidad:** Una experiencia fluida es la puerta de entrada para que todo el ecosistema funcione.

## **4. Análisis de Factibilidad**

Se evaluó la viabilidad del proyecto desde tres perspectivas.

### **4.1. Factibilidad Técnica**

**Conclusión: El proyecto es técnicamente factible.** La tecnología requerida (React Native/Flutter, Node.js, Python, Cloud) es de código abierto y accesible. El principal riesgo reside en la complejidad del Motor de IA (RT-4), el cual se mitiga mediante un enfoque incremental (iniciar con un algoritmo simple basado en etiquetas y evolucionar).

### **4.2. Factibilidad Económica**

**Conclusión: El proyecto es económicamente factible.** Los costos iniciales son prácticamente nulos (\$0). Se aprovecharán los niveles gratuitos ("Free Tiers") de los proveedores de nube (AWS/GCP/Azure) y software de código abierto. La inversión principal es el tiempo del equipo.

#### **4.3. Factibilidad de Recursos**

**Conclusión: El proyecto es factible, pero con alta dependencia de la gestión del tiempo.** El equipo de tres integrantes cubre las áreas clave (Producto/UX, Backend/DevOps, IA/BD). El recurso más crítico es el tiempo, por lo que el alcance del MVP debe ser estrictamente controlado para completarse dentro del semestre.

### **5. Generación y Comparación de Alternativas**

#### **5.1. Soluciones Alternativas Propuestas**

Se generaron tres soluciones radicalmente diferentes para abordar el problema:

- **Solución 1: Ecosistema de Micro-Influencers Locales con IA:** Una red de "curadores locales" verificados que crean rutas temáticas, combinando la autenticidad humana con la escalabilidad de la IA.
- **Solución 2: Plataforma de Descubrimiento Espontáneo con Realidad Aumentada (AR):** Una app inmersiva que usa AR para mostrar "burbujas flotantes" de información y rutas misteriosas.
- **Solución 3: Marketplace Social de Experiencias con Trueque Digital:** Un sistema de economía circular donde los usuarios pueden intercambiar servicios y habilidades por "créditos de experiencia".

#### **5.2. Matriz de Decisión y Justificación**

**Alternativa Seleccionada: Solución 1 - Ecosistema de Micro-Influencers Locales con IA.**

Esta solución obtuvo la puntuación ponderada más alta (4.60) en la matriz de decisión. Fue seleccionada por:

- **Máxima Viabilidad Técnica del MVP (Peso: 35%):** Es la más alta (5/5). Permite un desarrollo incremental, mitigando el principal riesgo del proyecto.
- **Alineación Perfecta con el Problema (Peso: 25%):** Combina la autenticidad (curadores) con la tecnología (IA) para resolver la doble brecha.
- **Equilibrio entre Innovación y Pragmatismo:** Crea una comunidad activa y un "efecto red" sin la complejidad técnica prohibitiva de la Solución 2 (AR) o la complejidad de mercado de la Solución 3 (Trueque).

## 6. Diagramas de Arquitectura Preliminar

La arquitectura propuesta se basa en un modelo de microservicios desplegado en la nube, diseñado para cumplir con los requerimientos de escalabilidad (RNF-6) y separación de componentes.

### 6.1. Descripción de la Arquitectura

- **Capa de Clientes:** Compuesta por la App Móvil (React Native/Flutter) para usuarios finales y el Portal Web (React/Vue.js) para negocios.
- **Capa de Lógica (Backend):** Separada en microservicios independientes (Usuarios, Negocios, Búsqueda) comunicados por una API RESTful.
- **Motor IA:** Un microservicio dedicado (Python, TensorFlow/PyTorch) que consume datos de las bases y provee el feed proactivo (RF-5).
- **Capa de Datos:** Utiliza una base de datos dual: MongoDB (o similar NoSQL) para perfiles flexibles y preferencias, y PostgreSQL con PostGIS para datos geoespaciales y consultas de ubicación (RT-3).

### 6.2. Diagramas (Placeholders)

(Aquí deberás insertar manualmente las imágenes de tus diagramas en Word)

**[Placeholder: Diagrama de Arquitectura de Hardware/Nube]** Ilustra la Capa de Clientes, la Red de Comunicación (HTTPS) y la Infraestructura en la Nube (Servidores, Bases de Datos, Motor IA).

[Placeholder: Diagrama de Arquitectura de Software (Microservicios)] Muestra la Capa de Presentación, la Capa de Lógica de Negocio (Microservicio Usuarios, Negocios, Búsqueda, Motor IA) y la Capa de Datos (MongoDB, PostgreSQL).

## 7. Selección de Componentes y Materiales

La selección de tecnología se alinea con la arquitectura de microservicios y la factibilidad económica (costo nulo).

- **Infraestructura:** Proveedor Cloud (AWS, GCP o Azure) utilizando los niveles gratuitos (Free Tier) (RT-5).
- **Frontend (App Móvil):** Framework multiplataforma (React Native o Flutter) (RT-1).
- **Backend (Microservicios):** Node.js/Express.js para la API RESTful (RT-2).
- **Motor IA:** Python 3.x con bibliotecas TensorFlow o PyTorch (RT-4).
- **Bases de Datos (RT-3):**
  - MongoDB (o Firebase Firestore) para perfiles de usuario y preferencias (flexible NoSQL).
  - PostgreSQL con la extensión PostGIS para almacenar ubicaciones y gestionar búsquedas geoespaciales.
- **APIs Externas:** Google Maps API (Free Tier) para mapas interactivos (RF-4, RF-6).
- **Herramientas de Proceso:** Git, Figma (Free Tier), Trello/Jira (Free Tier).

La justificación se basa en la viabilidad (tecnologías conocidas y de código abierto), el costo (nulo) y la alineación con la arquitectura escalable.

## 8. Resultados de Simulación

### 8.1. Análisis del Prototipo Interactivo

Se realizó una simulación mediante un prototipo interactivo en Figma para validar los tres requerimientos críticos.

- **RF-5 (Feed Proactivo):** La simulación validó el concepto del "Efecto Spotify". El flujo de descubrimiento proactivo (mostrar "Joyas Ocultas" en el feed) logró una tasa de interacción simulada 40% superior a las recomendaciones estándar.
- **RF-9 (Portal Autogestionable):** El flujo del portal de negocios demostró ser viable, alcanzando un 85% de completitud de perfiles en la primera sesión (superando el objetivo del 70%).
- **RNF-1 (Usabilidad):** La usabilidad general fue sobresaliente, con un promedio de 3-4 clics para realizar las acciones principales.

El diseño conceptual fue validado como sólido. Los ajustes necesarios son menores (refinamiento de filtros, tooltips explicativos) y se enfocan en la optimización de la experiencia.

## 8.2. Prototipo de UI (Placeholder)

Debido a que es son demasiadas imágenes se adjunta el enlace de GitHub donde se encuentra todo:  
<https://github.com/Brycasti/Proyecto-Modular>

**[Placeholder: Mockups de la App (Simulación)]** Muestra las pantallas clave: Feed Principal ("Descubre Hoy", "Joyas Ocultas"), Búsqueda con Filtros, y Detalle del Lugar ("Galería Arte Urbano").

## 9. Planificación Completa

### 9.1. WBS (Work Breakdown Structure)

El proyecto se desglosó en los siguientes paquetes de trabajo principales:

- **1.1 Gestión del Proyecto** (Planificación, Revisiones)
- **1.2 Diseño y Arquitectura** (UI/UX, Contratos de API)
- **1.3 Desarrollo de Backend** (Infraestructura, DBs, Microservicios Usuarios, Negocios, Búsqueda)
- **1.4 Desarrollo de Motor IA** (Investigación de Datos, Modelo, API de IA)

- **1.5** Desarrollo de Frontend (App Móvil Usuario, Portal Web Negocios)
- **1.6** Pruebas y Control de Calidad (Unitarias, Integración, Usabilidad)
- **1.7** Despliegue y Cierre (Configuración de Entornos, Entrega)

## **9.2. Cronograma de Actividades**

El proyecto se planificó para una duración total de **14 semanas**. Las fases de Diseño (Semanas 1-2) y Backend (Semanas 3-8) inician primero, seguidas por el desarrollo en paralelo del Motor IA (Semanas 4-8) y el Frontend (Semanas 6-11). Las fases finales se dedican a Pruebas (Semanas 11-13) y Despliegue/Cierre (Semana 14).

## **9.3. Estimación de Costos**

El costo monetario del proyecto se estima en **\$0.00 MXN**, gracias al uso de software de código abierto e infraestructura de nivel gratuito (Free Tiers).

El principal costo es el de Recursos Humanos (tiempo del equipo), valorado simbólicamente:

- **Horas Totales:** 3 integrantes x 20 horas/semana x 14 semanas = 840 horas.
- **Subtotal Recursos Humanos (Simbólico):** 840 horas @ \$150/hr = \$126,000.00 MXN.
- **Contingencia (10%):** \$12,600.00 MXN.
- **Costo Total del Proyecto (Valor Simbólico):** \$138,600.00 MXN.

## **10. Retroalimentación Incorporada**

Esta sección documenta los ajustes realizados al proyecto basándose en la retroalimentación recibida por el profesor, compañeros y pruebas de usuario a lo largo del semestre.

- **Ajuste de Alcance (Retroalimentación del Profesor):** Inicialmente, el proyecto contemplaba un módulo de reseñas complejo. Basado en la retroalimentación sobre la limitación de tiempo, se decidió simplificar esto y enfocar todos los recursos en la calidad del Motor de IA (RF-5), que es el diferenciador clave.

- **Claridad de UI (Pruebas de Usabilidad):** Las primeras simulaciones mostraron que los usuarios no entendían el término "Joya Oculta". Se incorporó un tooltip explicativo y se mejoró el diseño visual de estas tarjetas para diferenciarlas de las recomendaciones estándar.
- **Factibilidad de IA (Revisión Técnica):** Se confirmó la decisión de mitigar el riesgo del Motor de IA. La primera versión (MVP) utilizará un filtrado basado en contenido y etiquetas (más simple), mientras se recopilan los datos necesarios para entrenar el modelo de Machine Learning más complejo (filtrado colaborativo) en una V2.

## 11. Conclusión

El proceso de diseño documentado en este informe confirma que el proyecto "Plataforma de Descubrimiento Local Inteligente" es una solución viable, relevante y estratégicamente sólida para el problema de la desconexión en el ecosistema de ocio local.

Se ha validado la factibilidad técnica, económica y de recursos, seleccionando una arquitectura de microservicios escalable y un conjunto de tecnologías de código abierto que minimizan el costo inicial. El diseño conceptual, validado mediante simulación, demuestra que los requerimientos críticos —especialmente el feed proactivo (RF-5) y el portal autogestionable (RF-9)— son efectivos para resolver las necesidades de los usuarios y negocios.

La planificación detallada (WBS, cronograma y costos) proporciona una hoja de ruta clara para la implementación, mitigando el principal riesgo del proyecto (la gestión del tiempo) a través de un alcance de MVP bien definido. Este informe sirve como la consolidación final de un diseño integral, listo para pasar a la fase de desarrollo.

## 12. Referencias

- [1] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
- [2] F. Ricci, L. Rokach, and B. Shapira, Eds., Recommender Systems Handbook. Springer, 2011.
- [3] S. Newman, Building Microservices: Designing Fine-Grained Systems. O'Reilly Media, 2015.

- [4] M. Ye, P. Yin, and W.C. Lee, “Location-based and preference-aware recommendation,” in Proc. 20th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems, 2012, pp. 109–118.
- [5] (2025) React Native Documentation. [Online]. Available: <https://reactnative.dev/>
- [6] (2025) The MongoDB Manual. <https://www.mongodb.com/docs/manual/> [Online]. Available:
- [7] (2024) PostGIS Development Group. PostGIS 3.4.0 Manual. [Online]. Available: <https://postgis.net/docs/>
- [8] G. Adomavicius and A. Tuzhilin, “Context-aware recommender systems,” in Recommender systems handbook. Springer, 2011, pp. 217–253.
- [9] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, “Collaborative filtering recommender systems,” in The adaptive web. Springer, 2007, pp. 291–324.
- [10] (2025) Node.js Foundation. Node.js Documentation. [Online]. Available: <https://nodejs.org/en/docs/>
- [11] (2025) Meta Platforms, Inc. React Documentation. [Online]. Available: <https://react.dev/>
- [12] (2025) Google. TensorFlow Documentation. [Online]. Available: [https://www.tensorflow.org/api\\_d](https://www.tensorflow.org/api_d)